C# EJERCICIOS

NIVEL BÁSICO

LUIS MINGUILLÓN PASCUAL

C# ejercicios

Nivel basico

Luis Minguillón Pascual

COPYRICHT

La licencia de este libro electrónico es para uso personal. Por lo tanto no puedes revenderlo a otras personal.

Gracias por respetar los derechos del autor.

Luis Minguillón Pascual
©29/08/2025

Prefacio

La programación es una puerta de entrada a un mundo fascinante, lleno de posibilidades para crear, aprender y transformar ideas en realidad. Con solo unas líneas de código es posible construir herramientas, resolver problemas y dar forma a proyectos que antes parecían inalcanzables. C# es un lenguaje que, por su claridad y versatilidad, se convierte en un excelente punto de partida para quienes desean dar sus primeros pasos en este universo.

He decidido crear este libro pensando en los principiantes, en quienes se acercan por primera vez a la programación y buscan una manera sencilla y comprensible de aprender. Cada ejercicio está diseñado para acompañar el proceso de forma gradual, facilitando la comprensión de los conceptos básicos y ofreciendo una base sólida para avanzar hacia niveles más altos de conocimiento.

No puedo dejar de expresar mi más sincero agradecimiento a Marisa, mi mujer, cuyo apoyo incondicional durante la creación de este libro ha sido fundamental. Su paciencia, ánimo y confianza me dieron la fuerza necesaria para llevar a cabo este proyecto.

Con estas páginas espero que encuentres no solo ejercicios, sino también la motivación para descubrir y disfrutar el apasionante camino de la programación.

Luis Minguillón Pascual

Indice

C# Aplicaciones Consola

- **1. Hola Mundo**
- **2. Suma de Dos Números**
- **3. Calculadora Básica**
- **4. Número Par o Impar**
- **5. Factorial de un Número**
- **6. Tabla de Multiplicar**
- **7. Número Mayor de Tres**
- **8. Contador de Dígitos**
- **9. Números del 1 al 100**
- **10. Suma de Números Pares**
- **11. Conversión de Temperatura**
- **12. Área de un Círculo**
- **13. Números Primos**
- **14. Invertir Número**
- **15. Palíndromo**
- **16. Fibonacci**
- **17. Calculadora de IMC**
- **18. Contador de Vocales**
- **19. Generador de Números Aleatorios**
- **20. Calculadora de Edad**
- **21. Conversor de Moneda**
- **22. Validación de Email Básica**
- **23. Calculadora de Descuento**
- **24. Contador de Palabras**
- **25. Números Ascendentes**
- **26. Números Descendentes**
- **27. Suma de Dígitos**
- **28. Máximo Común Divisor**
- **29. Mínimo Común Múltiplo**
- **30. Calculadora de Potencia**
- **31. Números Perfectos**
- **32. Generador de Patrón**
- **33. Calculadora de Promedio**
- **34. Contador de Letras**
- **35. Conversor de Tiempo**
- **36. Números Armstrong**
- **37. Calculadora de Interés Simple**
- **38. Generador de Contraseña Simple**
- **39. Calculadora de Área Triángulo**
- **40. Verificador de Año Bisiesto**
- **41. Contador de Números Positivos y Negativos**
- **42. Calculadora de Volumen Esfera**

- **43. Conversor de Binario a Decimal**
- **44. Generador de Tablas de Multiplicar**
- **45. Calculadora de Perímetro Rectángulo**
- **46. Verificador de Número Perfecto**
- **47. Calculadora de Desviación Estándar**
- **48. Generador de Números Primos**
- **49. Calculadora de Hipotenusa**
- **50. Conversor de Decimal a Binario**
- **51. Calculadora de Factorial con Recursión**
- **52. Generador de Secuencia Collatz**
- **53. Calculadora de Media Armónica**
- **54. Verificador de Número Capicúa**
- **55. Generador de Números Romanos**
- **56. Calculadora de Combinaciones**
- **57. Generador de Matriz Identidad**
- **58. Calculadora de Ecuación Cuadrática**
- **59. Conversor de Minúsculas a Mayúsculas**
- **60. Generador de Patrón Numérico**
- **61. Calculadora de Área Círculo**
- **62. Verificador de Palíndromo Numérico**
- **63. Generador de Números Aleatorios en Rango**
- **64. Calculadora de Perímetro Círculo**
- **65. Contador de Consonantes**
- **66. Generador de Serie Geométrica**
- **67. Calculadora de IMC con Categoría**
- **68. Conversor de Segundos a Días**
- **69. Generador de Números Pares**
- **70. Calculadora de Velocidad**
- **71. Verificador de Número Positivo**
- **72. Generador de Números Impares**
- **73. Calculadora de Densidad**
- **74. Conversor de Minutos a Horas**
- **75. Generador de Patrón de Letras**
- **76. Calculadora de Fuerza**
- **77. Verificador de Divisibilidad**
- **78. Generador de Números Cuadrados**
- **79. Calculadora de Trabajo**
- **80. Programa de Despedida**

C# FORMS

- **Programa 1: Hola Mundo**
- **Programa 2: Suma de Dos Números**
- **Programa 3: Calculadora Básica**
- **Programa 4: Conversor Celsius a Fahrenheit**
- **Programa 5: Contador de Clicks**

- **Programa 6: Cambiar Color de Fondo**
- **Programa 7: Mostrar/Ocultar Texto**
- **Programa 8: Calculadora de Edad**
- **Programa 9: Lista de Tareas Simple**
- **Programa 10: Generador de Números Aleatorios**
- **Programa 11: Conversor de Texto a Mayúsculas**
- **Programa 12: Reloj Digital**
- **Programa 13: Calculadora de IMC**
- **Programa 14: Selector de Color**
- **Programa 15: Validación de Email Simple**
- **Programa 16: Contador de Palabras**
- **Programa 17: Calculadora de Propina**
- **Programa 18: Selector de Fuente**
- **Programa 19: Calculadora de Descuento**
- **Programa 20: Reproductor de Sonido Simple**
- **Programa 21: Calculadora de Área de Rectángulo**
- **Programa 22: Verificador de Número Par/Impar**
- **Programa 23: Conversor de Moneda (Dólares a Euros)**
- **Programa 24: Temporizador con ProgressBar**
- **Programa 25: Selector de Imagen**
- **Programa 26: Calculadora de Interés Simple**
- **Programa 27: Generador de Tabla de Multiplicar**
- **Programa 28: Contador Regresivo**
- **Programa 29: Calculadora de Promedio**
- **Programa 30: Selector de Fecha con MessageBox**
- **Programa 31: Conversor de Minúsculas a Mayúsculas**
- **Programa 32: Calculadora de Potencia**
- **Programa 33: Selector de Color para Texto**
- **Programa 34: Calculadora de Factorial**
- **Programa 35: Verificador de Palíndromo**
- **Programa 36: Calculadora de Descuento en Porcentaje**
- **Programa 37: Selector de Tamaño de Fuente**
- **Programa 38: Calculadora de Velocidad**
- **Programa 39: Selector de Idioma con RadioButtons**
- **Programa 40: Calculadora de Área de Círculo**
- **Programa 41: Contador de Caracteres**
- **Programa 42: Selector de Día de la Semana**
- **Programa 43: Calculadora de IVA**
- **Programa 44: Verificador de Año Bisiesto**
- **Programa 45: Selector de Múltiple con CheckBox**
- **Programa 46: Calculadora de Hipotenusa**
- **Programa 47: Generador de Números Primos**
- **Programa 48: Selector de Rango con TrackBar**
- **Programa 49: Calculadora de Días entre Fechas**
- **Programa 50: Conversor de Segundos a Minutos y Horas**
- **Programa 51: Verificador de Contraseña Simple**

- **Programa 52: Selector de Color con RGB**
- **Programa 53: Calculadora de Volumen de Esfera**
- **Programa 54: Selector de Tamaño de Formulario**
- **Programa 55: Generador de Números de Fibonacci**
- **Programa 56: Selector de Opacity del Formulario**
- **Programa 57: Calculadora de Desviación Estándar**
- **Programa 58: Selector de Sonido del Sistema**
- **Programa 59: Calculadora de Edad en Días**
- **Programa 60: Selector de FormBorderStyle**
- **Programa 61: Verificador de Número Primo**
- **Programa 62: Selector de Cursor**
- **Programa 63: Calculadora de Área de Triángulo**
- **Programa 64: Generador de Contraseña Aleatoria**
- **Programa 65: Selector de TransparencyKey**
- **Programa 66: Calculadora de MCD (Máximo Común Divisor)**
- **Programa 67: Selector de StartPosition**
- **Programa 68: Conversor de Sistema Decimal a Binario**
- **Programa 69: Selector de TopMost**
- **Programa 70: Calculadora de Media, Mediana y Moda**
- **Programa 71: Selector de Icono del Formulario**
- **Programa 72: Generador de Tabla ASCII**
- **Programa 73: Selector de ShowInTaskbar**
- **Programa 74: Calculadora de Ecuación Cuadrática**
- **Programa 75: Selector de MinimizeBox y MaximizeBox**
- **Programa 76: Conversor de Temperatura Completo**
- **Programa 77: Selector de WindowState**
- **Programa 78: Generador de Números Aleatorios con Rango**
- **Programa 79: Selector de ControlBox**
- **Programa 80: Juego de Adivinar el Número**

C# aplicaciones consola

80 Programas en C# Console nivel principiante

1. Hola Mundo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.WriteLine("¡Hola Mundo!");
        Console.ReadKey();
    }
}
```
Solución: El programa más básico que imprime un mensaje en la consola.

2. Suma de Dos Números
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese primer número: ");
        int num1 = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese segundo número: ");
        int num2 = Convert.ToInt32(Console.ReadLine());

        int suma = num1 + num2;
        Console.WriteLine($"La suma es: {suma}");
        Console.ReadKey();
    }
}
```
Solución: Solicita dos números al usuario y muestra su suma.

3. Calculadora Básica
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese primer número: ");
        double num1 = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese operador (+, -, *, /): ");
```

```
        char operador = Convert.ToChar(Console.ReadLine());

        Console.Write("Ingrese segundo número: ");
        double num2 = Convert.ToDouble(Console.ReadLine());

        double resultado = 0;

        switch (operador)
        {
            case '+': resultado = num1 + num2; break;
            case '-': resultado = num1 - num2; break;
            case '*': resultado = num1 * num2; break;
            case '/': resultado = num1 / num2; break;
            default: Console.WriteLine("Operador inválido"); break;
        }

        Console.WriteLine($"Resultado: {resultado}");
        Console.ReadKey();
    }
}
```
Solución: Realiza operaciones matemáticas básicas.

4. Número Par o Impar
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());

        if (numero % 2 == 0)
            Console.WriteLine("El número es par");
        else
            Console.WriteLine("El número es impar");

        Console.ReadKey();
    }
}
```
Solución: Determina si un número es par o impar usando el operador módulo.

5. Factorial de un Número
```csharp
using System;

class Program
{
    static void Main()
```

```
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int factorial = 1;

        for (int i = 1; i <= numero; i++)
        {
            factorial *= i;
        }

        Console.WriteLine($"Factorial de {numero} es: {factorial}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el factorial de un número usando un bucle for.

6. Tabla de Multiplicar
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());

        Console.WriteLine($"Tabla de multiplicar del {numero}:");
        for (int i = 1; i <= 10; i++)
        {
            Console.WriteLine($"{numero} x {i} = {numero * i}");
        }

        Console.ReadKey();
    }
}
```
Solución: Muestra la tabla de multiplicar de un número.

7. Número Mayor de Tres
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese primer número: ");
        int num1 = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese segundo número: ");
```

```
        int num2 = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese tercer número: ");
        int num3 = Convert.ToInt32(Console.ReadLine());

        int mayor = Math.Max(num1, Math.Max(num2, num3));
        Console.WriteLine($"El número mayor es: {mayor}");
        Console.ReadKey();
    }
}
```
Solución: Encuentra el número mayor entre tres números.

8. Contador de Dígitos
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int contador = 0;

        while (numero != 0)
        {
            numero /= 10;
            contador++;
        }

        Console.WriteLine($"El número tiene {contador} dígitos");
        Console.ReadKey();
    }
}
```
Solución: Cuenta la cantidad de dígitos de un número.

9. Números del 1 al 100
```csharp
using System;

class Program
{
    static void Main()
    {
        for (int i = 1; i <= 100; i++)
        {
            Console.Write(i + " ");
        }
        Console.ReadKey();
    }
```

```
}
```
Solución: Imprime los números del 1 al 100.

10. Suma de Números Pares
```csharp
using System;

class Program
{
    static void Main()
    {
        int suma = 0;
        for (int i = 1; i <= 100; i++)
        {
            if (i % 2 == 0)
                suma += i;
        }

        Console.WriteLine($"Suma de números pares del 1 al 100: {suma}");
        Console.ReadKey();
    }
}
```
Solución: Calcula la suma de todos los números pares entre 1 y 100.

11. Conversión de Temperatura
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese temperatura en Celsius: ");
        double celsius = Convert.ToDouble(Console.ReadLine());

        double fahrenheit = (celsius * 9/5) + 32;
        Console.WriteLine($"{celsius}°C = {fahrenheit}°F");
        Console.ReadKey();
    }
}
```
Solución: Convierte temperatura de Celsius a Fahrenheit.

12. Área de un Círculo
```csharp
using System;

class Program
{
    static void Main()
```

```
    {
        Console.Write("Ingrese el radio del círculo: ");
        double radio = Convert.ToDouble(Console.ReadLine());

        double area = Math.PI * Math.Pow(radio, 2);
        Console.WriteLine($"El área del círculo es: {area:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el área de un círculo dado su radio.

13. Números Primos
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        bool esPrimo = true;

        if (numero <= 1) esPrimo = false;
        else
        {
            for (int i = 2; i <= Math.Sqrt(numero); i++)
            {
                if (numero % i == 0)
                {
                    esPrimo = false;
                    break;
                }
            }
        }

        Console.WriteLine(esPrimo ? "Es primo" : "No es primo");
        Console.ReadKey();
    }
}
```
Solución: Verifica si un número es primo.

14. Invertir Número
```csharp
using System;

class Program
{
    static void Main()
    {
```

```
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int invertido = 0;

        while (numero != 0)
        {
            invertido = invertido * 10 + numero % 10;
            numero /= 10;
        }

        Console.WriteLine($"Número invertido: {invertido}");
        Console.ReadKey();
    }
}
```
Solución: Invierte los dígitos de un número.

15. Palíndromo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese una palabra: ");
        string palabra = Console.ReadLine();

        string invertida = "";
        for (int i = palabra.Length - 1; i >= 0; i--)
        {
            invertida += palabra[i];
        }

        if (palabra.Equals(invertida, StringComparison.OrdinalIgnoreCase))
            Console.WriteLine("Es palíndromo");
        else
            Console.WriteLine("No es palíndromo");

        Console.ReadKey();
    }
}
```
Solución: Verifica si una palabra es palíndromo.

16. Fibonacci
```csharp
using System;

class Program
{
    static void Main()
```

```
    {
        Console.Write("Ingrese cantidad de términos: ");
        int n = Convert.ToInt32(Console.ReadLine());

        int a = 0, b = 1, c;
        Console.Write("Serie Fibonacci: " + a + " " + b + " ");

        for (int i = 2; i < n; i++)
        {
            c = a + b;
            Console.Write(c + " ");
            a = b;
            b = c;
        }

        Console.ReadKey();
    }
}
```
Solución: Genera la serie Fibonacci.

17. Calculadora de IMC
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese peso (kg): ");
        double peso = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese altura (m): ");
        double altura = Convert.ToDouble(Console.ReadLine());

        double imc = peso / (altura * altura);
        Console.WriteLine($"Su IMC es: {imc:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el Índice de Masa Corporal.

18. Contador de Vocales
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese una frase: ");
```

```
        string frase = Console.ReadLine().ToLower();
        int contador = 0;

        foreach (char c in frase)
        {
            if ("aeiou".Contains(c))
                contador++;
        }

        Console.WriteLine($"Número de vocales: {contador}");
        Console.ReadKey();
    }
}
```
Solución: Cuenta las vocales en una frase.

19. Generador de Números Aleatorios
```csharp
using System;

class Program
{
    static void Main()
    {
        Random random = new Random();
        Console.Write("¿Cuántos números aleatorios desea? ");
        int cantidad = Convert.ToInt32(Console.ReadLine());

        Console.Write("Números aleatorios: ");
        for (int i = 0; i < cantidad; i++)
        {
            Console.Write(random.Next(1, 101) + " ");
        }

        Console.ReadKey();
    }
}
```
Solución: Genera números aleatorios.

20. Calculadora de Edad
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese su año de nacimiento: ");
        int añoNacimiento = Convert.ToInt32(Console.ReadLine());
        int añoActual = DateTime.Now.Year;
```

```
        int edad = añoActual - añoNacimiento;
        Console.WriteLine($"Su edad es: {edad} años");
        Console.ReadKey();
    }
}
```
Solución: Calcula la edad basada en el año de nacimiento.

21. Conversor de Moneda
```csharp
using System;

class Program
{
    static void Main()
    {
        const double TIPO_CAMBIO = 18.5; // Ejemplo: pesos por dólar
        Console.Write("Ingrese cantidad en dólares: ");
        double dolares = Convert.ToDouble(Console.ReadLine());

        double pesos = dolares * TIPO_CAMBIO;
        Console.WriteLine($"{dolares} dólares = {pesos} pesos");
        Console.ReadKey();
    }
}
```
Solución: Convierte dólares a pesos.

22. Validación de Email Básica
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese email: ");
        string email = Console.ReadLine();

        if (email.Contains("@") && email.Contains("."))
            Console.WriteLine("Email válido");
        else
            Console.WriteLine("Email inválido");

        Console.ReadKey();
    }
}
```
Solución: Validación básica de email.

23. Calculadora de Descuento

```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese precio original: ");
        double precio = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese porcentaje de descuento: ");
        double descuento = Convert.ToDouble(Console.ReadLine());

        double precioFinal = precio - (precio * descuento / 100);
        Console.WriteLine($"Precio final: {precioFinal:C}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el precio con descuento.

24. Contador de Palabras
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese una frase: ");
        string frase = Console.ReadLine();

        string[] palabras = frase.Split(' ', StringSplitOptions.RemoveEmptyEntries);
        Console.WriteLine($"Número de palabras: {palabras.Length}");
        Console.ReadKey();
    }
}
```
Solución: Cuenta las palabras en una frase.

25. Números Ascendentes
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese cantidad de números: ");
        int n = Convert.ToInt32(Console.ReadLine());

        for (int i = 1; i <= n; i++)
```

```
        {
            Console.Write(i + " ");
        }
        Console.ReadKey();
    }
}
```
Solución: Imprime números en orden ascendente.

26. Números Descendentes
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número inicial: ");
        int n = Convert.ToInt32(Console.ReadLine());

        for (int i = n; i >= 1; i--)
        {
            Console.Write(i + " ");
        }
        Console.ReadKey();
    }
}
```
Solución: Imprime números en orden descendente.

27. Suma de Dígitos
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int suma = 0;

        while (numero != 0)
        {
            suma += numero % 10;
            numero /= 10;
        }

        Console.WriteLine($"Suma de dígitos: {suma}");
        Console.ReadKey();
    }
}
```

```
**Solución:** Suma los dígitos de un número.

## 28. Máximo Común Divisor
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese primer número: ");
        int a = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese segundo número: ");
        int b = Convert.ToInt32(Console.ReadLine());

        while (b != 0)
        {
            int temp = b;
            b = a % b;
            a = temp;
        }

        Console.WriteLine($"MCD: {a}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el Máximo Común Divisor.

29. Mínimo Común Múltiplo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese primer número: ");
        int a = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese segundo número: ");
        int b = Convert.ToInt32(Console.ReadLine());

        int mcd = a;
        int temp = b;

        while (temp != 0)
        {
            int t = temp;
            temp = mcd % temp;
```
```

```
            mcd = t;
        }

        int mcm = (a * b) / mcd;
        Console.WriteLine($"MCM: {mcm}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el Mínimo Común Múltiplo.

30. Calculadora de Potencia
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese base: ");
        double baseNum = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese exponente: ");
        double exponente = Convert.ToDouble(Console.ReadLine());

        double resultado = Math.Pow(baseNum, exponente);
        Console.WriteLine($"Resultado: {resultado}");
        Console.ReadKey();
    }
}
```
Solución: Calcula la potencia de un número.

31. Números Perfectos
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int suma = 0;

        for (int i = 1; i < numero; i++)
        {
            if (numero % i == 0)
                suma += i;
        }

        if (suma == numero)
```

```
            Console.WriteLine("Es número perfecto");
        else
            Console.WriteLine("No es número perfecto");

        Console.ReadKey();
    }
}
```
Solución: Verifica si un número es perfecto.

32. Generador de Patrón
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número de filas: ");
        int filas = Convert.ToInt32(Console.ReadLine());

        for (int i = 1; i <= filas; i++)
        {
            for (int j = 1; j <= i; j++)
            {
                Console.Write("* ");
            }
            Console.WriteLine();
        }
        Console.ReadKey();
    }
}
```
Solución: Genera un patrón de asteriscos.

33. Calculadora de Promedio
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("¿Cuántos números desea promediar? ");
        int n = Convert.ToInt32(Console.ReadLine());
        double suma = 0;

        for (int i = 0; i < n; i++)
        {
            Console.Write($"Ingrese número {i + 1}: ");
            suma += Convert.ToDouble(Console.ReadLine());
        }
```

```
        double promedio = suma / n;
        Console.WriteLine($"Promedio: {promedio:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el promedio de n números.

34. Contador de Letras
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese una palabra: ");
        string palabra = Console.ReadLine();

        Console.WriteLine($"La palabra tiene {palabra.Length} letras");
        Console.ReadKey();
    }
}
```
Solución: Cuenta las letras de una palabra.

35. Conversor de Tiempo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese segundos: ");
        int segundos = Convert.ToInt32(Console.ReadLine());

        int horas = segundos / 3600;
        int minutos = (segundos % 3600) / 60;
        int segs = segundos % 60;

        Console.WriteLine($"{horas}h {minutos}m {segs}s");
        Console.ReadKey();
    }
}
```
Solución: Convierte segundos a horas, minutos y segundos.

36. Números Armstrong
```csharp
using System;
```

```
class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int temp = numero;
        int suma = 0;
        int digitos = numero.ToString().Length;

        while (temp != 0)
        {
            int digito = temp % 10;
            suma += (int)Math.Pow(digito, digitos);
            temp /= 10;
        }

        if (suma == numero)
            Console.WriteLine("Es número Armstrong");
        else
            Console.WriteLine("No es número Armstrong");

        Console.ReadKey();
    }
}
```
Solución: Verifica si un número es Armstrong.

37. Calculadora de Interés Simple
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese capital: ");
        double capital = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese tasa de interés anual (%): ");
        double tasa = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese tiempo en años: ");
        double tiempo = Convert.ToDouble(Console.ReadLine());

        double interes = (capital * tasa * tiempo) / 100;
        Console.WriteLine($"Interés simple: {interes:C}");
        Console.ReadKey();
    }
}
```

Solución: Calcula el interés simple.

38. Generador de Contraseña Simple
```csharp
using System;

class Program
{
    static void Main()
    {
        Random random = new Random();
        string caracteres =
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789";
        string contraseña = "";

        Console.Write("Longitud de contraseña: ");
        int longitud = Convert.ToInt32(Console.ReadLine());

        for (int i = 0; i < longitud; i++)
        {
            contraseña += caracteres[random.Next(caracteres.Length)];
        }

        Console.WriteLine($"Contraseña generada: {contraseña}");
        Console.ReadKey();
    }
}
```
Solución: Genera una contraseña aleatoria.

39. Calculadora de Área Triángulo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese base del triángulo: ");
        double baseTri = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese altura del triángulo: ");
        double altura = Convert.ToDouble(Console.ReadLine());

        double area = (baseTri * altura) / 2;
        Console.WriteLine($"Área del triángulo: {area}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el área de un triángulo.

40. Verificador de Año Bisiesto
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un año: ");
        int año = Convert.ToInt32(Console.ReadLine());

        if ((año % 4 == 0 && año % 100 != 0) || año % 400 == 0)
            Console.WriteLine("Es año bisiesto");
        else
            Console.WriteLine("No es año bisiesto");

        Console.ReadKey();
    }
}
```
Solución: Verifica si un año es bisiesto.

41. Contador de Números Positivos y Negativos
```csharp
using System;

class Program
{
    static void Main()
    {
        int positivos = 0, negativos = 0, ceros = 0;

        Console.Write("¿Cuántos números desea ingresar? ");
        int n = Convert.ToInt32(Console.ReadLine());

        for (int i = 0; i < n; i++)
        {
            Console.Write($"Número {i + 1}: ");
            double num = Convert.ToDouble(Console.ReadLine());

            if (num > 0) positivos++;
            else if (num < 0) negativos++;
            else ceros++;
        }

        Console.WriteLine($"Positivos: {positivos}, Negativos: {negativos}, Ceros: {ceros}");
        Console.ReadKey();
    }
}
```
Solución: Cuenta números positivos, negativos y ceros.

42. Calculadora de Volumen Esfera
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese radio de la esfera: ");
        double radio = Convert.ToDouble(Console.ReadLine());

        double volumen = (4.0 / 3.0) * Math.PI * Math.Pow(radio, 3);
        Console.WriteLine($"Volumen de la esfera: {volumen:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el volumen de una esfera.

43. Conversor de Binario a Decimal
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número binario: ");
        string binario = Console.ReadLine();

        int decimal = 0;
        for (int i = 0; i < binario.Length; i++)
        {
            if (binario[i] == '1')
                decimal += (int)Math.Pow(2, binario.Length - 1 - i);
        }

        Console.WriteLine($"Decimal: {decimal}");
        Console.ReadKey();
    }
}
```
Solución: Convierte binario a decimal.

44. Generador de Tablas de Multiplicar
```csharp
using System;

class Program
{
    static void Main()
    {
```

```
        for (int i = 1; i <= 10; i++)
        {
            Console.WriteLine($"\nTabla del {i}:");
            for (int j = 1; j <= 10; j++)
            {
                Console.WriteLine($"{i} x {j} = {i * j}");
            }
        }
        Console.ReadKey();
    }
}
```
Solución: Genera todas las tablas de multiplicar del 1 al 10.

45. Calculadora de Perímetro Rectángulo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese largo: ");
        double largo = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese ancho: ");
        double ancho = Convert.ToDouble(Console.ReadLine());

        double perimetro = 2 * (largo + ancho);
        Console.WriteLine($"Perímetro del rectángulo: {perimetro}");
        Console.ReadKey();
    }
}
```
Solución: Calcula el perímetro de un rectángulo.

46. Verificador de Número Perfecto
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int suma = 0;

        for (int i = 1; i <= numero / 2; i++)
        {
            if (numero % i == 0)
                suma += i;
```

```
        }

        if (suma == numero)
            Console.WriteLine("Es número perfecto");
        else
            Console.WriteLine("No es número perfecto");

        Console.ReadKey();
    }
}
```
Solución: Verifica si un número es perfecto.

47. Calculadora de Desviación Estándar
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("¿Cuántos números? ");
        int n = Convert.ToInt32(Console.ReadLine());
        double[] numeros = new double[n];
        double suma = 0;

        for (int i = 0; i < n; i++)
        {
            Console.Write($"Número {i + 1}: ");
            numeros[i] = Convert.ToDouble(Console.ReadLine());
            suma += numeros[i];
        }

        double media = suma / n;
        double sumaCuadrados = 0;

        foreach (double num in numeros)
        {
            sumaCuadrados += Math.Pow(num - media, 2);
        }

        double desviacion = Math.Sqrt(sumaCuadrados / n);
        Console.WriteLine($"Desviación estándar: {desviacion:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula la desviación estándar.

48. Generador de Números Primos
```csharp
using System;
```

```
class Program
{
   static void Main()
   {
      Console.Write("Ingrese límite: ");
      int limite = Convert.ToInt32(Console.ReadLine());

      Console.WriteLine("Números primos:");
      for (int i = 2; i <= limite; i++)
      {
         bool esPrimo = true;
         for (int j = 2; j <= Math.Sqrt(i); j++)
         {
            if (i % j == 0)
            {
               esPrimo = false;
               break;
            }
         }
         if (esPrimo) Console.Write(i + " ");
      }
      Console.ReadKey();
   }
}
```
Solución: Genera números primos hasta un límite.

49. Calculadora de Hipotenusa
```csharp
using System;

class Program
{
   static void Main()
   {
      Console.Write("Ingrese cateto a: ");
      double a = Convert.ToDouble(Console.ReadLine());

      Console.Write("Ingrese cateto b: ");
      double b = Convert.ToDouble(Console.ReadLine());

      double hipotenusa = Math.Sqrt(a * a + b * b);
      Console.WriteLine($"Hipotenusa: {hipotenusa:F2}");
      Console.ReadKey();
   }
}
```
Solución: Calcula la hipotenusa de un triángulo rectángulo.

50. Conversor de Decimal a Binario
```csharp
```

```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número decimal: ");
        int decimal = Convert.ToInt32(Console.ReadLine());
        string binario = "";

        while (decimal > 0)
        {
            binario = (decimal % 2) + binario;
            decimal /= 2;
        }

        Console.WriteLine($"Binario: {binario}");
        Console.ReadKey();
    }
}
```
Solución: Convierte decimal a binario.

51. Calculadora de Factorial con Recursión
```csharp
using System;

class Program
{
    static long Factorial(int n)
    {
        if (n == 0) return 1;
        return n * Factorial(n - 1);
    }

    static void Main()
    {
        Console.Write("Ingrese un número: ");
        int numero = Convert.ToInt32(Console.ReadLine());

        Console.WriteLine($"Factorial: {Factorial(numero)}");
        Console.ReadKey();
    }
}
```
Solución: Calcula factorial usando recursión.

52. Generador de Secuencia Collatz
```csharp
using System;

class Program
```

```
{
   static void Main()
   {
      Console.Write("Ingrese un número: ");
      int numero = Convert.ToInt32(Console.ReadLine());

      Console.Write("Secuencia: " + numero + " ");
      while (numero != 1)
      {
         if (numero % 2 == 0)
            numero /= 2;
         else
            numero = numero * 3 + 1;
         Console.Write(numero + " ");
      }
      Console.ReadKey();
   }
}
```
Solución: Genera la secuencia de Collatz.

53. Calculadora de Media Armónica
```csharp
using System;

class Program
{
   static void Main()
   {
      Console.Write("¿Cuántos números? ");
      int n = Convert.ToInt32(Console.ReadLine());
      double sumaReciprocos = 0;

      for (int i = 0; i < n; i++)
      {
         Console.Write($"Número {i + 1}: ");
         double num = Convert.ToDouble(Console.ReadLine());
         sumaReciprocos += 1 / num;
      }

      double mediaArmonica = n / sumaReciprocos;
      Console.WriteLine($"Media armónica: {mediaArmonica:F2}");
      Console.ReadKey();
   }
}
```
Solución: Calcula la media armónica.

54. Verificador de Número Capicúa
```csharp
using System;
```

```
class Program
{
   static void Main()
   {
      Console.Write("Ingrese un número: ");
      int numero = Convert.ToInt32(Console.ReadLine());
      int original = numero;
      int invertido = 0;

      while (numero != 0)
      {
         invertido = invertido * 10 + numero % 10;
         numero /= 10;
      }

      if (original == invertido)
         Console.WriteLine("Es capicúa");
      else
         Console.WriteLine("No es capicúa");

      Console.ReadKey();
   }
}
```
Solución: Verifica si un número es capicúa.

55. Generador de Números Romanos
```csharp
using System;

class Program
{
   static void Main()
   {
      Console.Write("Ingrese número (1-10): ");
      int numero = Convert.ToInt32(Console.ReadLine());
      string romano = "";

      string[] romanos = {"", "I", "II", "III", "IV", "V", "VI", "VII", "VIII", "IX", "X"};

      if (numero >= 1 && numero <= 10)
         romano = romanos[numero];
      else
         romano = "Número no válido";

      Console.WriteLine($"Número romano: {romano}");
      Console.ReadKey();
   }
}
```
Solución: Convierte números a romanos (1-10).

56. Calculadora de Combinaciones
```csharp
using System;

class Program
{
    static long Factorial(int n)
    {
        long fact = 1;
        for (int i = 2; i <= n; i++)
            fact *= i;
        return fact;
    }

    static void Main()
    {
        Console.Write("Ingrese n: ");
        int n = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese r: ");
        int r = Convert.ToInt32(Console.ReadLine());

        long combinaciones = Factorial(n) / (Factorial(r) * Factorial(n - r));
        Console.WriteLine($"Combinaciones: {combinaciones}");
        Console.ReadKey();
    }
}
```
Solución: Calcula combinaciones matemáticas.

57. Generador de Matriz Identidad
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese tamaño de matriz: ");
        int n = Convert.ToInt32(Console.ReadLine());

        for (int i = 0; i < n; i++)
        {
            for (int j = 0; j < n; j++)
            {
                Console.Write(i == j ? "1 " : "0 ");
            }
            Console.WriteLine();
        }
        Console.ReadKey();
    }
}
```

```
**Solución:** Genera una matriz identidad.

## 58. Calculadora de Ecuación Cuadrática
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese a: ");
        double a = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese b: ");
        double b = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese c: ");
        double c = Convert.ToDouble(Console.ReadLine());

        double discriminante = b * b - 4 * a * c;

        if (discriminante > 0)
        {
            double x1 = (-b + Math.Sqrt(discriminante)) / (2 * a);
            double x2 = (-b - Math.Sqrt(discriminante)) / (2 * a);
            Console.WriteLine($"Soluciones: {x1:F2}, {x2:F2}");
        }
        else if (discriminante == 0)
        {
            double x = -b / (2 * a);
            Console.WriteLine($"Solución única: {x:F2}");
        }
        else
        {
            Console.WriteLine("No hay soluciones reales");
        }
        Console.ReadKey();
    }
}
```
Solución: Resuelve ecuaciones cuadráticas.

59. Conversor de Minúsculas a Mayúsculas
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese texto: ");
```
```

```
        string texto = Console.ReadLine();

        Console.WriteLine($"En mayúsculas: {texto.ToUpper()}");
        Console.ReadKey();
    }
}
```
Solución: Convierte texto a mayúsculas.

60. Generador de Patrón Numérico
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número de filas: ");
        int filas = Convert.ToInt32(Console.ReadLine());

        for (int i = 1; i <= filas; i++)
        {
            for (int j = 1; j <= i; j++)
            {
                Console.Write(j + " ");
            }
            Console.WriteLine();
        }
        Console.ReadKey();
    }
}
```
Solución: Genera patrón numérico.

61. Calculadora de Área Círculo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese radio: ");
        double radio = Convert.ToDouble(Console.ReadLine());

        double area = Math.PI * radio * radio;
        Console.WriteLine($"Área: {area:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula área de círculo.

62. Verificador de Palíndromo Numérico
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número: ");
        int numero = Convert.ToInt32(Console.ReadLine());
        int original = numero;
        int invertido = 0;

        while (numero != 0)
        {
            invertido = invertido * 10 + numero % 10;
            numero /= 10;
        }

        Console.WriteLine(original == invertido ? "Es palíndromo" : "No es palíndromo");
        Console.ReadKey();
    }
}
```
Solución: Verifica palíndromos numéricos.

63. Generador de Números Aleatorios en Rango
```csharp
using System;

class Program
{
    static void Main()
    {
        Random random = new Random();
        Console.Write("Límite inferior: ");
        int min = Convert.ToInt32(Console.ReadLine());

        Console.Write("Límite superior: ");
        int max = Convert.ToInt32(Console.ReadLine());

        Console.Write("Cantidad: ");
        int cantidad = Convert.ToInt32(Console.ReadLine());

        Console.Write("Números aleatorios: ");
        for (int i = 0; i < cantidad; i++)
        {
            Console.Write(random.Next(min, max + 1) + " ");
        }
        Console.ReadKey();
    }
```

```
}
```
Solución: Genera números aleatorios en un rango.

64. Calculadora de Perímetro Círculo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese radio: ");
        double radio = Convert.ToDouble(Console.ReadLine());

        double perimetro = 2 * Math.PI * radio;
        Console.WriteLine($"Perímetro: {perimetro:F2}");
        Console.ReadKey();
    }
}
```
Solución: Calcula perímetro de círculo.

65. Contador de Consonantes
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese texto: ");
        string texto = Console.ReadLine().ToLower();
        int consonantes = 0;

        foreach (char c in texto)
        {
            if (char.IsLetter(c) && !"aeiou".Contains(c))
                consonantes++;
        }

        Console.WriteLine($"Consonantes: {consonantes}");
        Console.ReadKey();
    }
}
```
Solución: Cuenta consonantes en texto.

66. Generador de Serie Geométrica
```csharp
using System;
```

```
class Program
{
   static void Main()
   {
      Console.Write("Ingrese primer término: ");
      double a = Convert.ToDouble(Console.ReadLine());

      Console.Write("Ingrese razón: ");
      double r = Convert.ToDouble(Console.ReadLine());

      Console.Write("Ingrese número de términos: ");
      int n = Convert.ToInt32(Console.ReadLine());

      Console.Write("Serie geométrica: ");
      for (int i = 0; i < n; i++)
      {
         Console.Write(a * Math.Pow(r, i) + " ");
      }
      Console.ReadKey();
   }
}
```
Solución: Genera serie geométrica.

67. Calculadora de IMC con Categoría
```csharp
using System;

class Program
{
   static void Main()
   {
      Console.Write("Ingrese peso (kg): ");
      double peso = Convert.ToDouble(Console.ReadLine());

      Console.Write("Ingrese altura (m): ");
      double altura = Convert.ToDouble(Console.ReadLine());

      double imc = peso / (altura * altura);
      string categoria;

      if (imc < 18.5) categoria = "Bajo peso";
      else if (imc < 25) categoria = "Normal";
      else if (imc < 30) categoria = "Sobrepeso";
      else categoria = "Obesidad";

      Console.WriteLine($"IMC: {imc:F2} - {categoria}");
      Console.ReadKey();
   }
}
```
Solución: Calcula IMC con categoría.

68. Conversor de Segundos a Días
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese segundos: ");
        long segundos = Convert.ToInt64(Console.ReadLine());

        long dias = segundos / 86400;
        Console.WriteLine($"Días: {dias}");
        Console.ReadKey();
    }
}
```
Solución: Convierte segundos a días.

69. Generador de Números Pares
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese límite: ");
        int limite = Convert.ToInt32(Console.ReadLine());

        Console.Write("Números pares: ");
        for (int i = 2; i <= limite; i += 2)
        {
            Console.Write(i + " ");
        }
        Console.ReadKey();
    }
}
```
Solución: Genera números pares.

70. Calculadora de Velocidad
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese distancia (km): ");
        double distancia = Convert.ToDouble(Console.ReadLine());
```

```
        Console.Write("Ingrese tiempo (horas): ");
        double tiempo = Convert.ToDouble(Console.ReadLine());

        double velocidad = distancia / tiempo;
        Console.WriteLine($"Velocidad: {velocidad} km/h");
        Console.ReadKey();
    }
}
```
Solución: Calcula velocidad.

71. Verificador de Número Positivo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese número: ");
        double numero = Convert.ToDouble(Console.ReadLine());

        if (numero > 0)
            Console.WriteLine("Positivo");
        else if (numero < 0)
            Console.WriteLine("Negativo");
        else
            Console.WriteLine("Cero");

        Console.ReadKey();
    }
}
```
Solución: Verifica signo de número.

72. Generador de Números Impares
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese límite: ");
        int limite = Convert.ToInt32(Console.ReadLine());

        Console.Write("Números impares: ");
        for (int i = 1; i <= limite; i += 2)
        {
            Console.Write(i + " ");
        }
```

```
        Console.ReadKey();
    }
}
```
Solución: Genera números impares.

73. Calculadora de Densidad
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese masa (kg): ");
        double masa = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese volumen (m³): ");
        double volumen = Convert.ToDouble(Console.ReadLine());

        double densidad = masa / volumen;
        Console.WriteLine($"Densidad: {densidad} kg/m³");
        Console.ReadKey();
    }
}
```
Solución: Calcula densidad.

74. Conversor de Minutos a Horas
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese minutos: ");
        int minutos = Convert.ToInt32(Console.ReadLine());

        int horas = minutos / 60;
        int mins = minutos % 60;

        Console.WriteLine($"{horas}h {mins}m");
        Console.ReadKey();
    }
}
```
Solución: Convierte minutos a horas.

75. Generador de Patrón de Letras
```csharp
using System;
```

```
class Program
{
   static void Main()
   {
      Console.Write("Ingrese número de filas: ");
      int filas = Convert.ToInt32(Console.ReadLine());

      for (char c = 'A'; c < 'A' + filas; c++)
      {
         for (char d = 'A'; d <= c; d++)
         {
            Console.Write(d + " ");
         }
         Console.WriteLine();
      }
      Console.ReadKey();
   }
}
```
Solución: Genera patrón de letras.

76. Calculadora de Fuerza
```csharp
using System;

class Program
{
   static void Main()
   {
      Console.Write("Ingrese masa (kg): ");
      double masa = Convert.ToDouble(Console.ReadLine());

      Console.Write("Ingrese aceleración (m/s²): ");
      double aceleracion = Convert.ToDouble(Console.ReadLine());

      double fuerza = masa * aceleracion;
      Console.WriteLine($"Fuerza: {fuerza} N");
      Console.ReadKey();
   }
}
```
Solución: Calcula fuerza (F = m*a).

77. Verificador de Divisibilidad
```csharp
using System;

class Program
{
   static void Main()
   {
```

```
        Console.Write("Ingrese número: ");
        int numero = Convert.ToInt32(Console.ReadLine());

        Console.Write("Ingrese divisor: ");
        int divisor = Convert.ToInt32(Console.ReadLine());

        if (numero % divisor == 0)
            Console.WriteLine("Es divisible");
        else
            Console.WriteLine("No es divisible");

        Console.ReadKey();
    }
}
```
Solución: Verifica divisibilidad.

78. Generador de Números Cuadrados
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese límite: ");
        int limite = Convert.ToInt32(Console.ReadLine());

        Console.Write("Números cuadrados: ");
        for (int i = 1; i <= limite; i++)
        {
            Console.Write(i * i + " ");
        }
        Console.ReadKey();
    }
}
```
Solución: Genera números cuadrados.

79. Calculadora de Trabajo
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.Write("Ingrese fuerza (N): ");
        double fuerza = Convert.ToDouble(Console.ReadLine());

        Console.Write("Ingrese distancia (m): ");
        double distancia = Convert.ToDouble(Console.ReadLine());
```

```
        double trabajo = fuerza * distancia;
        Console.WriteLine($"Trabajo: {trabajo} J");
        Console.ReadKey();
    }
}
```
Solución: Calcula trabajo (W = F*d).

80. Programa de Despedida
```csharp
using System;

class Program
{
    static void Main()
    {
        Console.WriteLine("¡Gracias por usar estos programas!");
        Console.WriteLine("¡Hasta pronto!");
        Console.ReadKey();
    }
}
```
Solución: Programa final de despedida.

C# FORMS

80 Programas en C# FORM nivel principiante

Programa 1: Hola Mundo
Controles:
- Label (lblMensaje) - Text: ""
- Button (btnSaludar) - Text: "Saludar"

Código:
```csharp
private void btnSaludar_Click(object sender, EventArgs e)
{
    lblMensaje.Text = "¡Hola Mundo!";
}
```

Explicación: Al hacer clic en el botón, se muestra "¡Hola Mundo!" en la etiqueta.

Programa 2: Suma de Dos Números
Controles:
- 2 TextBox (txtNum1, txtNum2)
- Button (btnSumar) - Text: "Sumar"
- Label (lblResultado) - Text: "Resultado: "

Código:
```csharp
private void btnSumar_Click(object sender, EventArgs e)
{
    int num1 = int.Parse(txtNum1.Text);
    int num2 = int.Parse(txtNum2.Text);
    int resultado = num1 + num2;
    lblResultado.Text = $"Resultado: {resultado}";
}
```

Explicación: Toma dos números, los suma y muestra el resultado.

Programa 3: Calculadora Básica
Controles:
- 2 TextBox (txtNum1, txtNum2)
- 4 Button (btnSuma, btnResta, btnMultiplica, btnDivide)
- Label (lblResultado)

Código:
```csharp
```

```
private void btnSuma_Click(object sender, EventArgs e)
{
    double num1 = double.Parse(txtNum1.Text);
    double num2 = double.Parse(txtNum2.Text);
    lblResultado.Text = $"Resultado: {num1 + num2}";
}

private void btnResta_Click(object sender, EventArgs e)
{
    double num1 = double.Parse(txtNum1.Text);
    double num2 = double.Parse(txtNum2.Text);
    lblResultado.Text = $"Resultado: {num1 - num2}";
}
// Similar para multiplicación y división
```

Programa 4: Conversor Celsius a Fahrenheit

Controles:
- TextBox (txtCelsius)
- Button (btnConvertir) - Text: "Convertir"
- Label (lblFahrenheit)

Código:
```csharp
private void btnConvertir_Click(object sender, EventArgs e)
{
    double celsius = double.Parse(txtCelsius.Text);
    double fahrenheit = (celsius * 9/5) + 32;
    lblFahrenheit.Text = $"{fahrenheit} °F";
}
```

Programa 5: Contador de Clicks

Controles:
- Button (btnContar) - Text: "Click Me"
- Label (lblContador) - Text: "0"

Código:
```csharp
private int contador = 0;

private void btnContar_Click(object sender, EventArgs e)
{
    contador++;
    lblContador.Text = contador.ToString();
}
```

Programa 6: Cambiar Color de Fondo

Controles:
- 3 Button (btnRojo, btnVerde, btnAzul)
- Form (cambiar propiedad BackColor)

Código:
```csharp
private void btnRojo_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Red;
}

private void btnVerde_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Green;
}

private void btnAzul_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Blue;
}
```

Programa 7: Mostrar/Ocultar Texto

Controles:
- Label (lblTexto) - Text: "Texto Oculto"
- Button (btnToggle) - Text: "Mostrar/Ocultar"

Código:
```csharp
private void btnToggle_Click(object sender, EventArgs e)
{
    lblTexto.Visible = !lblTexto.Visible;
}
```

Programa 8: Calculadora de Edad

Controles:
- DateTimePicker (dtpFechaNacimiento)
- Button (btnCalcular) - Text: "Calcular Edad"
- Label (lblEdad)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
```

```
    DateTime fechaNacimiento = dtpFechaNacimiento.Value;
    int edad = DateTime.Now.Year - fechaNacimiento.Year;
    if (DateTime.Now.DayOfYear < fechaNacimiento.DayOfYear)
        edad--;
    lblEdad.Text = $"Edad: {edad} años";
}
```

Programa 9: Lista de Tareas Simple

Controles:
- TextBox (txtTarea)
- Button (btnAgregar) - Text: "Agregar"
- ListBox (lstTareas)

Código:
```csharp
private void btnAgregar_Click(object sender, EventArgs e)
{
    if (!string.IsNullOrEmpty(txtTarea.Text))
    {
        lstTareas.Items.Add(txtTarea.Text);
        txtTarea.Clear();
    }
}
```

Programa 10: Generador de Números Aleatorios

Controles:
- Button (btnGenerar) - Text: "Generar"
- Label (lblNumero)

Código:
```csharp
private Random random = new Random();

private void btnGenerar_Click(object sender, EventArgs e)
{
    int numero = random.Next(1, 101);
    lblNumero.Text = numero.ToString();
}
```

Programa 11: Conversor de Texto a Mayúsculas

Controles:
- TextBox (txtEntrada)
- Button (btnConvertir) - Text: "A Mayúsculas"

- Label (lblResultado)

Código:
```csharp
private void btnConvertir_Click(object sender, EventArgs e)
{
    lblResultado.Text = txtEntrada.Text.ToUpper();
}
```

Programa 12: Reloj Digital
Controles:
- Timer (timer1) - Interval: 1000
- Label (lblReloj)

Código:
```csharp
private void timer1_Tick(object sender, EventArgs e)
{
    lblReloj.Text = DateTime.Now.ToString("HH:mm:ss");
}

private void Form1_Load(object sender, EventArgs e)
{
    timer1.Start();
}
```

Programa 13: Calculadora de IMC
Controles:
- 2 TextBox (txtPeso, txtAltura)
- Button (btnCalcular) - Text: "Calcular IMC"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double peso = double.Parse(txtPeso.Text);
    double altura = double.Parse(txtAltura.Text);
    double imc = peso / (altura * altura);
    lblResultado.Text = $"IMC: {imc:F2}";
}
```

Programa 14: Selector de Color

Controles:
- ColorDialog (colorDialog1)
- Button (btnSeleccionarColor) - Text: "Seleccionar Color"
- Panel (panelColor)

Código:
```csharp
private void btnSeleccionarColor_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    {
        panelColor.BackColor = colorDialog1.Color;
    }
}
```

Programa 15: Validación de Email Simple
Controles:
- TextBox (txtEmail)
- Button (btnValidar) - Text: "Validar Email"
- Label (lblResultado)

Código:
```csharp
private void btnValidar_Click(object sender, EventArgs e)
{
    string email = txtEmail.Text;
    if (email.Contains("@") && email.Contains("."))
    {
        lblResultado.Text = "Email válido";
        lblResultado.ForeColor = Color.Green;
    }
    else
    {
        lblResultado.Text = "Email inválido";
        lblResultado.ForeColor = Color.Red;
    }
}
```

Programa 16: Contador de Palabras
Controles:
- TextBox (txtTexto) - Multiline: true
- Button (btnContar) - Text: "Contar Palabras"
- Label (lblContador)

Código:
```csharp
```

```
private void btnContar_Click(object sender, EventArgs e)
{
    string texto = txtTexto.Text;
    int contador = texto.Split(new char[] { ' ', '\n', '\r' },
                        StringSplitOptions.RemoveEmptyEntries).Length;
    lblContador.Text = $"Palabras: {contador}";
}
```

Programa 17: Calculadora de Propina

Controles:
- 2 TextBox (txtTotal, txtPorcentaje)
- Button (btnCalcular) - Text: "Calcular Propina"
- Label (lblPropina)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double total = double.Parse(txtTotal.Text);
    double porcentaje = double.Parse(txtPorcentaje.Text);
    double propina = total * (porcentaje / 100);
    lblPropina.Text = $"Propina: {propina:C}";
}
```

Programa 18: Selector de Fuente

Controles:
- FontDialog (fontDialog1)
- Button (btnSeleccionarFuente) - Text: "Seleccionar Fuente"
- Label (lblTexto)

Código:
```csharp
private void btnSeleccionarFuente_Click(object sender, EventArgs e)
{
    if (fontDialog1.ShowDialog() == DialogResult.OK)
    {
        lblTexto.Font = fontDialog1.Font;
    }
}
```

Programa 19: Calculadora de Descuento

Controles:
- 2 TextBox (txtPrecio, txtDescuento)

- Button (btnCalcular) - Text: "Calcular"
- Label (lblPrecioFinal)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double precio = double.Parse(txtPrecio.Text);
    double descuento = double.Parse(txtDescuento.Text);
    double precioFinal = precio - (precio * descuento / 100);
    lblPrecioFinal.Text = $"Precio Final: {precioFinal:C}";
}
```

Programa 20: Reproductor de Sonido Simple
Controles:
- Button (btnReproducir) - Text: "Reproducir Sonido"
- SoundPlayer (soundPlayer1)

Código:
```csharp
private void btnReproducir_Click(object sender, EventArgs e)
{
    System.Media.SystemSounds.Beep.Play();
}
```

Programa 21: Calculadora de Área de Rectángulo
Controles:
- 2 TextBox (txtBase, txtAltura)
- Button (btnCalcular) - Text: "Calcular Área"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double baseRect = double.Parse(txtBase.Text);
    double altura = double.Parse(txtAltura.Text);
    double area = baseRect * altura;
    lblResultado.Text = $"Área: {area:F2}";
}
```

Explicación: Calcula el área de un rectángulo multiplicando base por altura.

Programa 22: Verificador de Número Par/Impar
Controles:
- TextBox (txtNumero)
- Button (btnVerificar) - Text: "Verificar"
- Label (lblResultado)

Código:
```csharp
private void btnVerificar_Click(object sender, EventArgs e)
{
    int numero = int.Parse(txtNumero.Text);
    if (numero % 2 == 0)
        lblResultado.Text = "El número es PAR";
    else
        lblResultado.Text = "El número es IMPAR";
}
```

Explicación: Verifica si un número es par o impar usando el operador módulo.

Programa 23: Conversor de Moneda (Dólares a Euros)
Controles:
- TextBox (txtDolares)
- Button (btnConvertir) - Text: "Convertir a Euros"
- Label (lblEuros)

Código:
```csharp
private const double TIPO_CAMBIO = 0.85;

private void btnConvertir_Click(object sender, EventArgs e)
{
    double dolares = double.Parse(txtDolares.Text);
    double euros = dolares * TIPO_CAMBIO;
    lblEuros.Text = $"{euros:F2} €";
}
```

Explicación: Convierte dólares a euros usando un tipo de cambio fijo.

Programa 24: Temporizador con ProgressBar
Controles:
- ProgressBar (progressBar1) - Maximum: 100, Value: 0
- Timer (timer1) - Interval: 100
- Button (btnIniciar) - Text: "Iniciar"

Código:

```csharp
private void btnIniciar_Click(object sender, EventArgs e)
{
    progressBar1.Value = 0;
    timer1.Start();
}

private void timer1_Tick(object sender, EventArgs e)
{
    if (progressBar1.Value < 100)
        progressBar1.Value += 1;
    else
        timer1.Stop();
}
```

Explicación: Llena una barra de progreso gradualmente usando un temporizador.

Programa 25: Selector de Imagen

Controles:
- PictureBox (pictureBox1) - SizeMode: Zoom
- Button (btnSeleccionar) - Text: "Seleccionar Imagen"
- OpenFileDialog (openFileDialog1)

Código:
```csharp
private void btnSeleccionar_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "Imágenes|*.jpg;*.png;*.bmp";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        pictureBox1.Image = Image.FromFile(openFileDialog1.FileName);
    }
}
```

Explicación: Permite al usuario seleccionar y mostrar una imagen.

Programa 26: Calculadora de Interés Simple

Controles:
- 3 TextBox (txtCapital, txtTasa, txtTiempo)
- Button (btnCalcular) - Text: "Calcular Interés"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
```

```
    double capital = double.Parse(txtCapital.Text);
    double tasa = double.Parse(txtTasa.Text);
    double tiempo = double.Parse(txtTiempo.Text);
    double interes = capital * (tasa / 100) * tiempo;
    lblResultado.Text = $"Interés: {interes:C}";
}
```

Explicación: Calcula el interés simple usando la fórmula I = C * R * T.

Programa 27: Generador de Tabla de Multiplicar

Controles:
- TextBox (txtNumero)
- Button (btnGenerar) - Text: "Generar Tabla"
- ListBox (lstTabla)

Código:
```csharp
private void btnGenerar_Click(object sender, EventArgs e)
{
    lstTabla.Items.Clear();
    int numero = int.Parse(txtNumero.Text);
    for (int i = 1; i <= 10; i++)
    {
        lstTabla.Items.Add($"{numero} x {i} = {numero * i}");
    }
}
```

Explicación: Genera la tabla de multiplicar del número ingresado.

Programa 28: Contador Regresivo

Controles:
- TextBox (txtSegundos)
- Button (btnIniciar) - Text: "Iniciar"
- Label (lblCuentaRegresiva)
- Timer (timer1) - Interval: 1000

Código:
```csharp
private int segundosRestantes;

private void btnIniciar_Click(object sender, EventArgs e)
{
    segundosRestantes = int.Parse(txtSegundos.Text);
    lblCuentaRegresiva.Text = segundosRestantes.ToString();
    timer1.Start();
}
```

```
private void timer1_Tick(object sender, EventArgs e)
{
    segundosRestantes--;
    lblCuentaRegresiva.Text = segundosRestantes.ToString();

    if (segundosRestantes <= 0)
        timer1.Stop();
}
```

Explicación: Inicia una cuenta regresiva desde el número ingresado.

Programa 29: Calculadora de Promedio

Controles:
- 3 TextBox (txtNota1, txtNota2, txtNota3)
- Button (btnCalcular) - Text: "Calcular Promedio"
- Label (lblPromedio)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double nota1 = double.Parse(txtNota1.Text);
    double nota2 = double.Parse(txtNota2.Text);
    double nota3 = double.Parse(txtNota3.Text);
    double promedio = (nota1 + nota2 + nota3) / 3;
    lblPromedio.Text = $"Promedio: {promedio:F2}";
}
```

Explicación: Calcula el promedio de tres números.

Programa 30: Selector de Fecha con MessageBox

Controles:
- DateTimePicker (dtpFecha)
- Button (btnMostrar) - Text: "Mostrar Fecha"

Código:
```csharp
private void btnMostrar_Click(object sender, EventArgs e)
{
    MessageBox.Show($"Fecha seleccionada: {dtpFecha.Value.ToShortDateString()}",
                    "Fecha", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
```

Explicación: Muestra la fecha seleccionada en un cuadro de mensaje.

Programa 31: Conversor de Minúsculas a Mayúsculas
Controles:
- TextBox (txtTexto)
- Button (btnConvertir) - Text: "Convertir"
- Label (lblResultado)

Código:
```csharp
private void btnConvertir_Click(object sender, EventArgs e)
{
    lblResultado.Text = txtTexto.Text.ToUpper();
}
```

Explicación: Convierte texto a mayúsculas.

Programa 32: Calculadora de Potencia
Controles:
- 2 TextBox (txtBase, txtExponente)
- Button (btnCalcular) - Text: "Calcular Potencia"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double baseNum = double.Parse(txtBase.Text);
    double exponente = double.Parse(txtExponente.Text);
    double resultado = Math.Pow(baseNum, exponente);
    lblResultado.Text = $"Resultado: {resultado}";
}
```

Explicación: Calcula la potencia de un número.

Programa 33: Selector de Color para Texto
Controles:
- Label (lblTexto) - Text: "Texto de ejemplo"
- Button (btnCambiarColor) - Text: "Cambiar Color"
- ColorDialog (colorDialog1)

Código:
```csharp
private void btnCambiarColor_Click(object sender, EventArgs e)
{
```

```
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    {
        lblTexto.ForeColor = colorDialog1.Color;
    }
}
```

Explicación: Permite cambiar el color del texto de una etiqueta.

Programa 34: Calculadora de Factorial

Controles:
- TextBox (txtNumero)
- Button (btnCalcular) - Text: "Calcular Factorial"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    int numero = int.Parse(txtNumero.Text);
    long factorial = 1;

    for (int i = 1; i <= numero; i++)
    {
        factorial *= i;
    }

    lblResultado.Text = $"Factorial: {factorial}";
}
```

Explicación: Calcula el factorial de un número.

Programa 35: Verificador de Palíndromo

Controles:
- TextBox (txtPalabra)
- Button (btnVerificar) - Text: "Verificar"
- Label (lblResultado)

Código:
```csharp
private void btnVerificar_Click(object sender, EventArgs e)
{
    string palabra = txtPalabra.Text.ToLower();
    char[] caracteres = palabra.ToCharArray();
    Array.Reverse(caracteres);
    string palabraInvertida = new string(caracteres);
```

```
    if (palabra == palabraInvertida)
        lblResultado.Text = "Es un palíndromo";
    else
        lblResultado.Text = "No es un palíndromo";
}
```

Explicación: Verifica si una palabra es un palíndromo (se lee igual al derecho y al revés).

Programa 36: Calculadora de Descuento en Porcentaje

Controles:
- 2 TextBox (txtPrecioOriginal, txtPrecioDescuento)
- Button (btnCalcular) - Text: "Calcular Descuento"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double precioOriginal = double.Parse(txtPrecioOriginal.Text);
    double precioDescuento = double.Parse(txtPrecioDescuento.Text);
    double descuento = ((precioOriginal - precioDescuento) / precioOriginal) * 100;
    lblResultado.Text = $"Descuento: {descuento:F2}%";
}
```

Explicación: Calcula el porcentaje de descuento aplicado.

Programa 37: Selector de Tamaño de Fuente

Controles:
- TrackBar (trackBar1) - Minimum: 8, Maximum: 72
- Label (lblTexto) - Text: "Texto de ejemplo"

Código:
```csharp
private void trackBar1_Scroll(object sender, EventArgs e)
{
    lblTexto.Font = new Font(lblTexto.Font.FontFamily, trackBar1.Value);
}
```

Explicación: Permite cambiar el tamaño de fuente arrastrando un control deslizante.

Programa 38: Calculadora de Velocidad

Controles:
- 2 TextBox (txtDistancia, txtTiempo)

- Button (btnCalcular) - Text: "Calcular Velocidad"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double distancia = double.Parse(txtDistancia.Text);
    double tiempo = double.Parse(txtTiempo.Text);
    double velocidad = distancia / tiempo;
    lblResultado.Text = $"Velocidad: {velocidad:F2} km/h";
}
```

Explicación: Calcula la velocidad basada en distancia y tiempo.

Programa 39: Selector de Idioma con RadioButtons
Controles:
- 3 RadioButton (rbEspanol, rbIngles, rbFrances) - Text: "Español", "Inglés", "Francés"
- Button (btnSaludar) - Text: "Saludar"
- Label (lblSaludo)

Código:
```csharp
private void btnSaludar_Click(object sender, EventArgs e)
{
    if (rbEspanol.Checked)
        lblSaludo.Text = "¡Hola!";
    else if (rbIngles.Checked)
        lblSaludo.Text = "Hello!";
    else if (rbFrances.Checked)
        lblSaludo.Text = "Bonjour!";
}
```

Explicación: Muestra un saludo en diferentes idiomas según la selección.

Programa 40: Calculadora de Área de Círculo
Controles:
- TextBox (txtRadio)
- Button (btnCalcular) - Text: "Calcular Área"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double radio = double.Parse(txtRadio.Text);
```

```
    double area = Math.PI * Math.Pow(radio, 2);
    lblResultado.Text = $"Área: {area:F2}";
}
```

Explicación: Calcula el área de un círculo usando π * r².

Programa 41: Contador de Caracteres
Controles:
- TextBox (txtTexto) - Multiline: true
- Label (lblContador) - Text: "Caracteres: 0"

Código:
```csharp
private void txtTexto_TextChanged(object sender, EventArgs e)
{
    lblContador.Text = $"Caracteres: {txtTexto.Text.Length}";
}
```

Explicación: Muestra en tiempo real la cantidad de caracteres escritos.

Programa 42: Selector de Día de la Semana
Controles:
- ComboBox (cmbDias) - Items: Lunes, Martes, Miércoles, Jueves, Viernes, Sábado, Domingo
- Button (btnSeleccionar) - Text: "Seleccionar"
- Label (lblResultado)

Código:
```csharp
private void btnSeleccionar_Click(object sender, EventArgs e)
{
    lblResultado.Text = $"Día seleccionado: {cmbDias.SelectedItem}";
}
```

Explicación: Muestra el día seleccionado en un ComboBox.

Programa 43: Calculadora de IVA
Controles:
- TextBox (txtPrecioSinIVA)
- Button (btnCalcular) - Text: "Calcular IVA"
- Label (lblResultado)

Código:
```csharp
```

```
private const double IVA = 0.21;

private void btnCalcular_Click(object sender, EventArgs e)
{
    double precioSinIVA = double.Parse(txtPrecioSinIVA.Text);
    double precioConIVA = precioSinIVA * (1 + IVA);
    lblResultado.Text = $"Precio con IVA: {precioConIVA:C}";
}
```

Explicación: Calcula el precio con IVA incluido.

Programa 44: Verificador de Año Bisiesto

Controles:
- TextBox (txtAnio)
- Button (btnVerificar) - Text: "Verificar"
- Label (lblResultado)

Código:
```csharp
private void btnVerificar_Click(object sender, EventArgs e)
{
    int anio = int.Parse(txtAnio.Text);
    bool esBisiesto = (anio % 4 == 0 && anio % 100 != 0) || (anio % 400 == 0);

    if (esBisiesto)
        lblResultado.Text = $"{anio} es un año bisiesto";
    else
        lblResultado.Text = $"{anio} no es un año bisiesto";
}
```

Explicación: Verifica si un año es bisiesto según las reglas del calendario gregoriano.

Programa 45: Selector de Múltiple con CheckBox

Controles:
- 3 CheckBox (chkOpcion1, chkOpcion2, chkOpcion3) - Text: "Opción 1", "Opción 2", "Opción 3"
- Button (btnMostrar) - Text: "Mostrar Selección"
- Label (lblResultado)

Código:
```csharp
private void btnMostrar_Click(object sender, EventArgs e)
{
    string seleccion = "";

    if (chkOpcion1.Checked) seleccion += "Opción 1, ";
    if (chkOpcion2.Checked) seleccion += "Opción 2, ";
```

```
    if (chkOpcion3.Checked) seleccion += "Opción 3, ";

    if (seleccion.Length > 0)
        seleccion = seleccion.Remove(seleccion.Length - 2);
    else
        seleccion = "Ninguna opción seleccionada";

    lblResultado.Text = $"Seleccionado: {seleccion}";
}
```

Explicación: Muestra las opciones seleccionadas mediante CheckBox.

Programa 46: Calculadora de Hipotenusa

Controles:
- 2 TextBox (txtCateto1, txtCateto2)
- Button (btnCalcular) - Text: "Calcular Hipotenusa"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double cateto1 = double.Parse(txtCateto1.Text);
    double cateto2 = double.Parse(txtCateto2.Text);
    double hipotenusa = Math.Sqrt(Math.Pow(cateto1, 2) + Math.Pow(cateto2, 2));
    lblResultado.Text = $"Hipotenusa: {hipotenusa:F2}";
}
```

Explicación: Calcula la hipotenusa de un triángulo rectángulo usando el teorema de Pitágoras.

Programa 47: Generador de Números Primos

Controles:
- TextBox (txtLimite)
- Button (btnGenerar) - Text: "Generar Primos"
- ListBox (lstPrimos)

Código:
```csharp
private void btnGenerar_Click(object sender, EventArgs e)
{
    lstPrimos.Items.Clear();
    int limite = int.Parse(txtLimite.Text);

    for (int numero = 2; numero <= limite; numero++)
    {
        bool esPrimo = true;
```

```
        for (int divisor = 2; divisor <= Math.Sqrt(numero); divisor++)
        {
            if (numero % divisor == 0)
            {
                esPrimo = false;
                break;
            }
        }

        if (esPrimo)
            lstPrimos.Items.Add(numero);
    }
}
```

Explicación: Genera todos los números primos hasta un límite dado.

Programa 48: Selector de Rango con TrackBar

Controles:
- TrackBar (trackBar1) - Minimum: 0, Maximum: 100
- Label (lblValor) - Text: "Valor: 0"

Código:
```csharp
private void trackBar1_Scroll(object sender, EventArgs e)
{
    lblValor.Text = $"Valor: {trackBar1.Value}";
}
```

Explicación: Muestra el valor actual de un control deslizante.

Programa 49: Calculadora de Días entre Fechas

Controles:
- 2 DateTimePicker (dtpFecha1, dtpFecha2)
- Button (btnCalcular) - Text: "Calcular Días"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    TimeSpan diferencia = dtpFecha2.Value - dtpFecha1.Value;
    lblResultado.Text = $"Días de diferencia: {diferencia.TotalDays:F0}";
}
```

Explicación: Calcula la diferencia en días entre dos fechas.

Programa 50: Conversor de Segundos a Minutos y Horas
Controles:
- TextBox (txtSegundos)
- Button (btnConvertir) - Text: "Convertir"
- Label (lblResultado)

Código:
```csharp
private void btnConvertir_Click(object sender, EventArgs e)
{
    int segundos = int.Parse(txtSegundos.Text);
    int horas = segundos / 3600;
    int minutos = (segundos % 3600) / 60;
    int segsRestantes = segundos % 60;

    lblResultado.Text = $"{horas}h {minutos}m {segsRestantes}s";
}
```

Explicación: Convierte segundos a horas, minutos y segundos.

Programa 51: Verificador de Contraseña Simple
Controles:
- TextBox (txtPassword) - PasswordChar: *
- Button (btnVerificar) - Text: "Verificar"
- Label (lblResultado)

Código:
```csharp
private const string PASSWORD_CORRECTA = "secreto";

private void btnVerificar_Click(object sender, EventArgs e)
{
    if (txtPassword.Text == PASSWORD_CORRECTA)
    {
        lblResultado.Text = "Contraseña correcta";
        lblResultado.ForeColor = Color.Green;
    }
    else
    {
        lblResultado.Text = "Contraseña incorrecta";
        lblResultado.ForeColor = Color.Red;
    }
}
```

Explicación: Verifica si la contraseña ingresada coincide con una contraseña predefinida.

Programa 52: Selector de Color con RGB
Controles:
- 3 TrackBar (tbRojo, tbVerde, tbAzul) - Minimum: 0, Maximum: 255
- Panel (panelColor)

Código:
```csharp
private void ActualizarColor()
{
    int rojo = tbRojo.Value;
    int verde = tbVerde.Value;
    int azul = tbAzul.Value;
    panelColor.BackColor = Color.FromArgb(rojo, verde, azul);
}

private void tbRojo_Scroll(object sender, EventArgs e) => ActualizarColor();
private void tbVerde_Scroll(object sender, EventArgs e) => ActualizarColor();
private void tbAzul_Scroll(object sender, EventArgs e) => ActualizarColor();
```

Explicación: Permite crear un color personalizado mezclando componentes RGB.

Programa 53: Calculadora de Volumen de Esfera
Controles:
- TextBox (txtRadio)
- Button (btnCalcular) - Text: "Calcular Volumen"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double radio = double.Parse(txtRadio.Text);
    double volumen = (4.0 / 3.0) * Math.PI * Math.Pow(radio, 3);
    lblResultado.Text = $"Volumen: {volumen:F2}";
}
```

Explicación: Calcula el volumen de una esfera usando la fórmula $(4/3)\pi r^3$.

Programa 54: Selector de Tamaño de Formulario
Controles:
- ComboBox (cmbTamanios) - Items: Pequeño, Mediano, Grande
- Button (btnAplicar) - Text: "Aplicar Tamaño"

Código:
```csharp
private void btnAplicar_Click(object sender, EventArgs e)
{
    switch (cmbTamanios.SelectedItem.ToString())
    {
        case "Pequeño":
            this.Size = new Size(300, 200);
            break;
        case "Mediano":
            this.Size = new Size(600, 400);
            break;
        case "Grande":
            this.Size = new Size(900, 600);
            break;
    }
}
```

Explicación: Cambia el tamaño del formulario según la selección.

Programa 55: Generador de Números de Fibonacci

Controles:
- TextBox (txtCantidad)
- Button (btnGenerar) - Text: "Generar Secuencia"
- ListBox (lstFibonacci)

Código:
```csharp
private void btnGenerar_Click(object sender, EventArgs e)
{
    lstFibonacci.Items.Clear();
    int cantidad = int.Parse(txtCantidad.Text);

    long a = 0, b = 1;

    if (cantidad >= 1) lstFibonacci.Items.Add(a);
    if (cantidad >= 2) lstFibonacci.Items.Add(b);

    for (int i = 3; i <= cantidad; i++)
    {
        long c = a + b;
        lstFibonacci.Items.Add(c);
        a = b;
        b = c;
    }
}
```

Explicación: Genera la secuencia de Fibonacci con la cantidad de números especificada.

Programa 56: Selector de Opacity del Formulario
Controles:
- TrackBar (trackBar1) - Minimum: 10, Maximum: 100, Value: 100
- Label (lblOpacidad) - Text: "Opacidad: 100%"

Código:
```csharp
private void trackBar1_Scroll(object sender, EventArgs e)
{
    this.Opacity = trackBar1.Value / 100.0;
    lblOpacidad.Text = $"Opacidad: {trackBar1.Value}%";
}
```

Explicación: Controla la opacidad (transparencia) del formulario.

Programa 57: Calculadora de Desviación Estándar
Controles:
- TextBox (txtNumeros) - Text: "1,2,3,4,5"
- Button (btnCalcular) - Text: "Calcular Desviación"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    string[] numerosStr = txtNumeros.Text.Split(',');
    double[] numeros = Array.ConvertAll(numerosStr, double.Parse);

    double promedio = numeros.Average();
    double sumaDiferencias = numeros.Sum(n => Math.Pow(n - promedio, 2));
    double desviacion = Math.Sqrt(sumaDiferencias / numeros.Length);

    lblResultado.Text = $"Desviación estándar: {desviacion:F4}";
}
```

Explicación: Calcula la desviación estándar de un conjunto de números.

Programa 58: Selector de Sonido del Sistema
Controles:
- ComboBox (cmbSonidos) - Items: Asterisk, Beep, Exclamation, Hand, Question
- Button (btnReproducir) - Text: "Reproducir Sonido"

Código:
```csharp
private void btnReproducir_Click(object sender, EventArgs e)
{
    switch (cmbSonidos.SelectedItem.ToString())
    {
        case "Asterisk":
            SystemSounds.Asterisk.Play();
            break;
        case "Beep":
            SystemSounds.Beep.Play();
            break;
        case "Exclamation":
            SystemSounds.Exclamation.Play();
            break;
        case "Hand":
            SystemSounds.Hand.Play();
            break;
        case "Question":
            SystemSounds.Question.Play();
            break;
    }
}
```

Explicación: Reproduce diferentes sonidos del sistema según la selección.

Programa 59: Calculadora de Edad en Días
Controles:
- DateTimePicker (dtpFechaNacimiento)
- Button (btnCalcular) - Text: "Calcular Días"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    TimeSpan diferencia = DateTime.Today - dtpFechaNacimiento.Value;
    lblResultado.Text = $"Días vividos: {diferencia.TotalDays:F0}";
}
```

Explicación: Calcula cuántos días han pasado desde la fecha de nacimiento.

Programa 60: Selector de FormBorderStyle
Controles:

- ComboBox (cmbEstilos) - Items: None, FixedSingle, Fixed3D, FixedDialog, Sizable, FixedToolWindow, SizableToolWindow
- Button (btnAplicar) - Text: "Aplicar Estilo"

Código:
```csharp
private void btnAplicar_Click(object sender, EventArgs e)
{
    string estilo = cmbEstilos.SelectedItem.ToString();
    this.FormBorderStyle = (FormBorderStyle)Enum.Parse(typeof(FormBorderStyle), estilo);
}
```

Explicación: Cambia el estilo del borde del formulario.

Programa 61: Verificador de Número Primo

Controles:
- TextBox (txtNumero)
- Button (btnVerificar) - Text: "Verificar Primo"
- Label (lblResultado)

Código:
```csharp
private void btnVerificar_Click(object sender, EventArgs e)
{
    int numero = int.Parse(txtNumero.Text);
    bool esPrimo = true;

    if (numero < 2)
        esPrimo = false;
    else
    {
        for (int i = 2; i <= Math.Sqrt(numero); i++)
        {
            if (numero % i == 0)
            {
                esPrimo = false;
                break;
            }
        }
    }

    lblResultado.Text = esPrimo ? "Es primo" : "No es primo";
}
```

Explicación: Verifica si un número es primo.

Programa 62: Selector de Cursor
Controles:
- ComboBox (cmbCursores) - Items: Arrow, Cross, Hand, IBeam, WaitCursor
- Button (btnAplicar) - Text: "Aplicar Cursor"

Código:
```csharp
private void btnAplicar_Click(object sender, EventArgs e)
{
    string cursor = cmbCursores.SelectedItem.ToString();
    this.Cursor = (Cursor)Enum.Parse(typeof(Cursors), cursor);
}
```

Explicación: Cambia el cursor del ratón dentro del formulario.

Programa 63: Calculadora de Área de Triángulo
Controles:
- 2 TextBox (txtBase, txtAltura)
- Button (btnCalcular) - Text: "Calcular Área"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double baseTri = double.Parse(txtBase.Text);
    double altura = double.Parse(txtAltura.Text);
    double area = (baseTri * altura) / 2;
    lblResultado.Text = $"Área: {area:F2}";
}
```

Explicación: Calcula el área de un triángulo.

Programa 64: Generador de Contraseña Aleatoria
Controles:
- NumericUpDown (nudLongitud) - Minimum: 4, Maximum: 32, Value: 8
- Button (btnGenerar) - Text: "Generar Contraseña"
- TextBox (txtPassword)

Código:
```csharp
private Random random = new Random();
private const string CARACTERES =
"ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789!@#$
%^&*";
```

```
private void btnGenerar_Click(object sender, EventArgs e)
{
    int longitud = (int)nudLongitud.Value;
    char[] password = new char[longitud];

    for (int i = 0; i < longitud; i++)
    {
        password[i] = CARACTERES[random.Next(CARACTERES.Length)];
    }

    txtPassword.Text = new string(password);
}
```

Explicación: Genera una contraseña aleatoria con la longitud especificada.

Programa 65: Selector de TransparencyKey

Controles:
- Button (btnSeleccionarColor) - Text: "Seleccionar Color"
- ColorDialog (colorDialog1)

Código:
```csharp
private void btnSeleccionarColor_Click(object sender, EventArgs e)
{
    if (colorDialog1.ShowDialog() == DialogResult.OK)
    {
        this.TransparencyKey = colorDialog1.Color;
        this.BackColor = colorDialog1.Color;
    }
}
```

Explicación: Hace que un color seleccionado se vuelva transparente en el formulario.

Programa 66: Calculadora de MCD (Máximo Común Divisor)

Controles:
- 2 TextBox (txtNum1, txtNum2)
- Button (btnCalcular) - Text: "Calcular MCD"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    int a = int.Parse(txtNum1.Text);
    int b = int.Parse(txtNum2.Text);
```

```
    while (b != 0)
    {
        int temp = b;
        b = a % b;
        a = temp;
    }

    lblResultado.Text = $"MCD: {a}";
}
```

Explicación: Calcula el máximo común divisor usando el algoritmo de Euclides.

Programa 67: Selector de StartPosition

Controles:
- ComboBox (cmbPosiciones) - Items: CenterScreen, Manual, WindowsDefaultLocation, WindowsDefaultBounds, CenterParent
- Button (btnAplicar) - Text: "Aplicar Posición"

Código:
```csharp
private void btnAplicar_Click(object sender, EventArgs e)
{
    string posicion = cmbPosiciones.SelectedItem.ToString();
    this.StartPosition = (FormStartPosition)Enum.Parse(typeof(FormStartPosition), posicion);
    this.Close();
    this.Show(); // Para aplicar el cambio
}
```

Explicación: Cambia la posición inicial del formulario al abrirlo.

Programa 68: Conversor de Sistema Decimal a Binario

Controles:
- TextBox (txtDecimal)
- Button (btnConvertir) - Text: "Convertir a Binario"
- Label (lblResultado)

Código:
```csharp
private void btnConvertir_Click(object sender, EventArgs e)
{
    int decimalNum = int.Parse(txtDecimal.Text);
    string binario = Convert.ToString(decimalNum, 2);
    lblResultado.Text = $"Binario: {binario}";
}
```

Explicación: Convierte un número decimal a su representación binaria.

Programa 69: Selector de TopMost
Controles:
- CheckBox (chkSiempreVisible) - Text: "Siempre visible"

Código:
```csharp
private void chkSiempreVisible_CheckedChanged(object sender, EventArgs e)
{
    this.TopMost = chkSiempreVisible.Checked;
}
```

Explicación: Hace que el formulario siempre esté visible sobre otras ventanas.

Programa 70: Calculadora de Media, Mediana y Moda
Controles:
- TextBox (txtNumeros) - Text: "1,2,3,4,5"
- Button (btnCalcular) - Text: "Calcular"
- 3 Label (lblMedia, lblMediana, lblModa)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    string[] numerosStr = txtNumeros.Text.Split(',');
    double[] numeros = Array.ConvertAll(numerosStr, double.Parse);
    Array.Sort(numeros);

    // Media
    double media = numeros.Average();
    lblMedia.Text = $"Media: {media:F2}";

    // Mediana
    double mediana;
    int medio = numeros.Length / 2;
    if (numeros.Length % 2 == 0)
        mediana = (numeros[medio - 1] + numeros[medio]) / 2.0;
    else
        mediana = numeros[medio];
    lblMediana.Text = $"Mediana: {mediana:F2}";

    // Moda
    var grupos = numeros.GroupBy(n => n);
    int maxFrecuencia = grupos.Max(g => g.Count());
    var modas = grupos.Where(g => g.Count() == maxFrecuencia).Select(g => g.Key);
    lblModa.Text = $"Moda: {string.Join(", ", modas)}";
```

}
```

**Explicación:** Calcula la media, mediana y moda de un conjunto de números.

---

## Programa 71: Selector de Icono del Formulario
**Controles:**
- Button (btnSeleccionarIcono) - Text: "Seleccionar Icono"
- OpenFileDialog (openFileDialog1) - Filter: "Iconos|*.ico"

**Código:**
```csharp
private void btnSeleccionarIcono_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "Iconos|*.ico";
    if (openFileDialog1.ShowDialog() == DialogResult.OK)
    {
        this.Icon = new Icon(openFileDialog1.FileName);
    }
}
```

Explicación: Permite cambiar el icono del formulario.

Programa 72: Generador de Tabla ASCII
Controles:
- Button (btnGenerar) - Text: "Generar Tabla ASCII"
- ListBox (lstASCII)

Código:
```csharp
private void btnGenerar_Click(object sender, EventArgs e)
{
    lstASCII.Items.Clear();
    for (int i = 32; i <= 126; i++)
    {
        lstASCII.Items.Add($"{i}: {(char)i}");
    }
}
```

Explicación: Genera una tabla con los caracteres ASCII imprimibles.

Programa 73: Selector de ShowInTaskbar
Controles:
- CheckBox (chkMostrarEnBarra) - Text: "Mostrar en barra de tareas"
```

Código:
```csharp
private void chkMostrarEnBarra_CheckedChanged(object sender, EventArgs e)
{
    this.ShowInTaskbar = chkMostrarEnBarra.Checked;
}
```

Explicación: Controla si el formulario aparece en la barra de tareas.

Programa 74: Calculadora de Ecuación Cuadrática

Controles:
- 3 TextBox (txtA, txtB, txtC)
- Button (btnCalcular) - Text: "Calcular Raíces"
- Label (lblResultado)

Código:
```csharp
private void btnCalcular_Click(object sender, EventArgs e)
{
    double a = double.Parse(txtA.Text);
    double b = double.Parse(txtB.Text);
    double c = double.Parse(txtC.Text);

    double discriminante = b * b - 4 * a * c;

    if (discriminante < 0)
    {
        lblResultado.Text = "No hay raíces reales";
    }
    else if (discriminante == 0)
    {
        double raiz = -b / (2 * a);
        lblResultado.Text = $"Raíz única: {raiz:F2}";
    }
    else
    {
        double raiz1 = (-b + Math.Sqrt(discriminante)) / (2 * a);
        double raiz2 = (-b - Math.Sqrt(discriminante)) / (2 * a);
        lblResultado.Text = $"Raíces: {raiz1:F2} y {raiz2:F2}";
    }
}
```

Explicación: Resuelve ecuaciones cuadráticas de la forma $ax^2 + bx + c = 0$.

Programa 75: Selector de MinimizeBox y MaximizeBox

Controles:
- 2 CheckBox (chkMinimizar, chkMaximizar) - Text: "Botón Minimizar", "Botón Maximizar"

Código:
```csharp
private void chkMinimizar_CheckedChanged(object sender, EventArgs e)
{
    this.MinimizeBox = chkMinimizar.Checked;
}

private void chkMaximizar_CheckedChanged(object sender, EventArgs e)
{
    this.MaximizeBox = chkMaximizar.Checked;
}
```

Explicación: Controla la visibilidad de los botones de minimizar y maximizar.

Programa 76: Conversor de Temperatura Completo
Controles:
- TextBox (txtTemperatura)
- 2 ComboBox (cmbDe, cmbA) - Items: Celsius, Fahrenheit, Kelvin
- Button (btnConvertir) - Text: "Convertir"
- Label (lblResultado)

Código:
```csharp
private void btnConvertir_Click(object sender, EventArgs e)
{
    double temperatura = double.Parse(txtTemperatura.Text);
    string de = cmbDe.SelectedItem.ToString();
    string a = cmbA.SelectedItem.ToString();

    // Convertir a Celsius primero
    double celsius = 0;

    switch (de)
    {
        case "Celsius": celsius = temperatura; break;
        case "Fahrenheit": celsius = (temperatura - 32) * 5/9; break;
        case "Kelvin": celsius = temperatura - 273.15; break;
    }

    // Convertir de Celsius a la unidad destino
    double resultado = 0;

    switch (a)
    {
        case "Celsius": resultado = celsius; break;
        case "Fahrenheit": resultado = (celsius * 9/5) + 32; break;
```

```
        case "Kelvin": resultado = celsius + 273.15; break;
    }

    lblResultado.Text = $"{resultado:F2} {a}";
}
```

Explicación: Convierte entre diferentes escalas de temperatura.

Programa 77: Selector de WindowState
Controles:
- ComboBox (cmbEstado) - Items: Normal, Minimized, Maximized
- Button (btnAplicar) - Text: "Aplicar Estado"

Código:
```csharp
private void btnAplicar_Click(object sender, EventArgs e)
{
    string estado = cmbEstado.SelectedItem.ToString();
    this.WindowState = (FormWindowState)Enum.Parse(typeof(FormWindowState), estado);
}
```

Explicación: Cambia el estado de la ventana (normal, minimizada, maximizada).

Programa 78: Generador de Números Aleatorios con Rango
Controles:
- 2 NumericUpDown (nudMinimo, nudMaximo)
- Button (btnGenerar) - Text: "Generar"
- Label (lblNumero)

Código:
```csharp
private Random random = new Random();

private void btnGenerar_Click(object sender, EventArgs e)
{
    int minimo = (int)nudMinimo.Value;
    int maximo = (int)nudMaximo.Value;
    int numero = random.Next(minimo, maximo + 1);
    lblNumero.Text = numero.ToString();
}
```

Explicación: Genera un número aleatorio dentro de un rango especificado.

Programa 79: Selector de ControlBox
Controles:
- CheckBox (chkControlBox) - Text: "Mostrar barra de título"

Código:
```csharp
private void chkControlBox_CheckedChanged(object sender, EventArgs e)
{
    this.ControlBox = chkControlBox.Checked;
}
```

Explicación: Controla la visibilidad de la barra de título del formulario.

Programa 80: Juego de Adivinar el Número
Controles:
- TextBox (txtAdivinanza)
- Button (btnAdivinar) - Text: "Adivinar"
- Label (lblMensaje)
- Button (btnNuevoJuego) - Text: "Nuevo Juego"

Código:
```csharp
private int numeroSecreto;
private int intentos;

private void IniciarNuevoJuego()
{
    Random random = new Random();
    numeroSecreto = random.Next(1, 101);
    intentos = 0;
    lblMensaje.Text = "Adivina un número entre 1 y 100";
    txtAdivinanza.Text = "";
}

private void btnNuevoJuego_Click(object sender, EventArgs e) => IniciarNuevoJuego();

private void btnAdivinar_Click(object sender, EventArgs e)
{
    int adivinanza = int.Parse(txtAdivinanza.Text);
    intentos++;

    if (adivinanza < numeroSecreto)
        lblMensaje.Text = "Demasiado bajo. Intenta again.";
    else if (adivinanza > numeroSecreto)
        lblMensaje.Text = "Demasiado alto. Intenta again.";
    else
        lblMensaje.Text = $"¡Correcto! Lo adivinaste en {intentos} intentos.";
}
```

private void Form1_Load(object sender, EventArgs e) => IniciarNuevoJuego();
```

**Explicación:** Juego simple donde el usuario debe adivinar un número secreto.

---
```

Epílogo

Llegar hasta aquí significa que has recorrido un camino importante en tu aprendizaje de C#. Paso a paso, ejercicio a ejercicio, has explorado las bases de un lenguaje que abre las puertas a infinitas posibilidades en el mundo de la programación. Tal vez al inicio todo parecía complicado, pero ahora cuentas con herramientas y conocimientos que te permitirán seguir avanzando con confianza.

Este libro no pretende ser un punto final, sino más bien un trampolín. Cada concepto que has practicado es solo el comienzo de un viaje más amplio en el que podrás desarrollar proyectos más complejos, aprender nuevas técnicas y, sobre todo, dar vida a tus propias ideas. La programación es un arte y una disciplina a la vez: requiere práctica, paciencia y creatividad, pero también ofrece la satisfacción única de ver cómo tus esfuerzos se transforman en resultados reales.

Espero que estos ejercicios te hayan servido no solo para comprender la lógica de C#, sino también para despertar tu curiosidad y tu deseo de seguir aprendiendo. Recuerda que cada error es una oportunidad de mejora y que cada logro, por pequeño que sea, es un paso más hacia convertirte en un verdadero programador.

Con gratitud y entusiasmo, cierro estas páginas animándote a continuar explorando, creando y creciendo en este maravilloso mundo de la programación.